

REMARKS

Claims 1-40 are pending in the application.

Claims 1-40 stand rejected.

Claims 41-43 have been added. Support for the new claims can be found on pages 6-12.

The Information Disclosure Statement that was originally submitted on November 18, 2004 is being resubmitted with this response.

*Rejection of Claims under 35 U.S.C. §102*

Claims 1-40 stand rejected under 35 U.S.C. §102(b) as being anticipated by Template Software's Workflow product. Applicants respectfully traverse this rejection.

The cited art fails to anticipate, teach, or suggest "identifying at least one software component of the computing environment; and automatically generating an electronic document containing an identification of each of the at least one software component, the electronic document comprising instructions for automatically reconstructing the computing environment on the first computing hardware or on other computing hardware," as recited in claim 1.

The Office Action attempts to equate the work item taught in the cited documentation with the "electronic document" of claim 1. See e.g. Office Action, pages 4 and 10. The Office Action also states that Template Software's Workflow product teaches "an electronic document containing an identification of each of the at least one software component." Particularly, the Office Action says that "the identification of a software component is referring to the name of TASK(s)." Office Action, p. 10. However, the Office Action has not cited any portion of the reference that explicitly or inherently teaches that a work item contains an identification of each of the at least one software component. Instead, the Office Action states that this feature is anticipated by "the different Ids" in Figure 3-1 of "Using the Workflow Development Environment." Office Action, p. 4.

Figure 3-1 shows the user interface of the Workflow Design Editor, in which a workflow is being designed. The interface shows a list of work items in a work items list box as well as a design hierarchy list box that lists tasks. Simply identifying work items and tasks in this interface does not teach or suggest an "electronic document" that contains both "an identification

of each of the at least one software component” and “instructions for automatically reconstructing the computing environment,” as recited in claim 1. Applicant notes that a “claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference.” *Verdegall Bros. V. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Accordingly, the cited art fails to anticipate claim 1 for at least this reason.

Furthermore, it would appear unlikely for a work item (or any other electronic document taught in the cited portions of Template Software’s documentation) to need to identify a “software component” and “instructions for automatically reconstructing the computing environment.” In particular, Template Software’s product is directed to workflow management, not to automatically reconstructing computing environments. Accordingly, there would be no need for any electronic document used in Template Software’s product to include “instructions for automatically reconstructing the computing environment.” Similarly, there is no need for a work item to include information identifying a task (which the Office Action attempts to equate with a software component), since the “Processors” and “Destroyers” commands (see p. 3-7 of “Using the Workflow Design Editor”) are already available to define which tasks can process and destroy work items of particular types.

Furthermore, the assertion that “a task is a software component” (Office Action, p. 10), and that therefore identifying a task anticipates identifying a software component, ignores certain teachings of the Template Software product documentation. In particular, while a task may be performed by a software component, a task may also be performed manually. Thus identification of a task does not inherently involve identification of a software component. Applicant notes that p. 2-5 of “Developing a WFT Workflow System” describes several tasks that are performed by human employees: “Tasks... are work activities... Roles... indicate the kind of worker who performs one or more tasks -- for example, Employee, Inventory clerk, or Technician.” Additionally, p. 2-7 describes how the WFT editors can be used to “create points at which work items leave the automated portions of the workflow system to enter manual tasks” (emphasis added). Applicant also notes that the Examiner’s characterization of “Applicant’s attempt to refer to a task as something other than a software component” as “very objectionable” (Office Action, p. 10) is inconsistent with the teachings of “Developing a WFT Workflow System.”

As noted briefly above, the cited portions of the Template Software document further fail to anticipate, teach, or suggest an “electronic document comprising instructions for automatically reconstructing the computing environment.” The Examiner cites Chapters 6 and 7 of “Using “ as well as Chapters 2 and 3 of WFT as teaching “the electronic document comprising instructions for automatically reconstructing the computing environment.” Office Action, p. 4. Specifically, the Examiner says that a workflow item that is used in Template Software’s Workflow product anticipates the electronic document of claim 1 because “upon receiving the work item the Applications perform specific tasks on them depending on the type of work item received the Application is configured.” Office Action, p. 4.

Applicant notes that having an application perform particular tasks on a work item depending upon the type of work item received does not mean that the work item includes “instructions for automatically reconstructing the computing environment.” It simply means that the application is configured to recognize the type of work item received and to perform tasks on that work item that are appropriate to that type of work item. For example, based on user selection of the “processors” and “destroyers” commands described on page 3-7 of “Using the Workflow Design Editor,” a Template Software user can specify which tasks process or destroy work items of a particular type as those work items progress through the workflow. Furthermore, simply performing different functions on different types of work items is clearly not the same as “reconstructing the computing environment.” Different functions can easily be performed within the same computing environment, and thus there is no need to “[reconstruct] a computing environment” in order to be able to perform different functions on work items.

Nevertheless, the Office Action states: “The work item being an object and logic can be considered to reconstruct the computing environment in the broadest reasonable interpretation.” The Office Action then cites Figure 6-1 of “Using the WFT Development Environment” and states that “each Requisition in a business is not the exact same. The work item (electronic document) when passed from one ROLE to another will perform a form of reconstruction. Different operations need to be performed based on the information.” Office Action, p. 11.

This interpretation simply does not comport with the claim language. For example, page 4 of the specification describes the term “computing environment” as including “an operating system and all the applications and documents installed on top of it, or a collection of networked

computing environments.” Thus, simply performing “different operations” clearly fails to teach or suggest “reconstructing a computing environment.”

Furthermore, Figure 6-1, which is relied on to teach this feature of claim 1, simply illustrates the user interface of the task editor, which “provides menus and tools for editing tasks, creating forms, associating forms with work items in tasks, and accessing the Form Editor to edit forms.” “Using the Workflow Development Environment, p. 6-6. No portion of the task editor user interface explicitly or inherently teaches “automatically generating an electronic document... comprising instructions for automatically reconstructing the computing environment on the first computing hardware or on other computing hardware” (emphasis added). Accordingly, claim 1 is further patentable over the cited art for at least this reason.

Claims 2-40 are patentable for similar reasons to those provided above.

Further with respect to claim 15, the cited art fails to anticipate, teach, or suggest:

- obtaining an electronic document that contains a description of a computing environment to be installed;

- reading the electronic document;

- identifying candidate computing hardware for installation of the computing environment;

- selecting the target computing hardware from the candidate computing hardware for installation of the computing environment; and

- installing at least one software component on the target computing hardware in accordance with data contained in the electronic document.

The Office Action simply refers to the rejection of claim 1 as supporting the rejection of claim 15. However, claim 15 includes several features, such as “identifying candidate hardware for installation of the computing environment” and “installing at least one software component on the target computing hardware in accordance with data contained in the electronic document,” that are not included in claim 1. No portions of the cited have been cited as teaching or suggesting these features of claim 15. Accordingly, Applicant respectfully requests the withdrawal of the rejection of claim 15.

Added Claims

The cited art fails to anticipate, teach, or suggest "accessing an electronic document, wherein the electronic document comprises: information identifying one or more components of a computing environment and constraint information identifying a constraint associated with one of the one or more components; selecting hardware that satisfies the constraint identified in the electronic document; and installing the one or more components identified in the electronic document on the selected hardware," as recited in claim 41. Claims 41-43 are patentable over the cited art for at least this reason.

CONCLUSION

In view of the amendments and remarks set forth herein, the application is believed to be in condition for allowance and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the Examiner is invited to telephone the undersigned at 512-439-5087.

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Mail Stop Amendment, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on March 22, 2005.

*Brenna A. Brock*

Attorney for Applicants

*3-22-2005*

Date of Signature

Respectfully submitted,

*Brenna A. Brock*

Brenna A. Brock

Attorney for Applicants

Reg. No. 48,509

Telephone: (512) 439-5087

Facsimile: (512) 439-5099